## REMARKS

Applicant acknowledges receipt of the Advisory Action dated July 2, 2004. The Advisory Action indicates that Applicant's request for reconsideration dated June 1, 2004, does not place the application in condition for allowance. The Advisory Action attaches a continuation sheet which outlines the Examiner's rationale for maintaining the Final Rejection. Applicant respectfully requests the Examiner's reconsideration in light of the following remarks.

Applicant's June 1, 2004, reply to Final Office Action argued that claim 53 is distinguishable over Birrittella. Claim 53 provides:

> A method comprising:
> indicating in a configuration register that a vector buffer is in use, wherein said vector buffer is one of a plurality of vector buffers comprising a vector buffer pool; and
> transferring vector data between memory and a register file via said vector buffer, wherein said configuration register, said register file and said vector buffer pool are local to a processor, and said transferring is initiated by a first program;
> generating a vector transfer unit exception if a second program attempts to transfer vector data between memory and a register of the register file via any vector buffer of the vector buffer pool, if said configuration register indicates that said vector buffer is in use.

Independent claim 53 recites first and second programs. The first program transfers vector data between memory and a register file via said vector buffer. The second program attempts to transfer vector data between memory and a register. Birrittella does not teach or fairly suggest first and second programs as recited in independent claim 53.

The Advisory Action argues that instructions, not programs, access the vector buffer, presumably as a premise for additional arguments contained in the advisory

action. However, if the Examiner interprets independent claim 3 to read that instructions of a first program transfer vector data between memory and a register file, and that instructions of a second program attempts to transfer vector data between memory and a register of the register file, the Applicant asserts that claim 53 is nonetheless distinguishable over Birrittella since Birrittella does not teach generating a vector transfer unit exception if instructions of a second program attempts to transfer vector data.

The Advisory Action argues that since programs consist of instructions, the existence of instructions indicates the existence of programs. Applicant asserts that this is faulty logic. Since a program consists of instructions, it cannot always be said that the existence of instructions indicates the existence of programs.

Finally, the Advisory Action argues that installing all instructions currently executing within a processor would result in a dead lock situation where the processor would grind to a complete halt. Stall is defined in Hennessy, J.L. and D. A. Patterson, "Computer Architecture: A Quantitative Approach," Morgan Kaufmann Publishers, Inc.; 1990, page 214. More particularly, this reference defines, "A stall is where an instruction must pause one or more clock cycles waiting for some resources to be available." Stallling all instructions currently executing does not necessarily result in a deadlock situation where the processor would grind to a complete halt. Stalling all instructions currently executing may result in a pause of one or more clock cycles waiting for some resource to become available. Indeed, pausing to wait for the availability of resources appears to be the focus of Birrittella. More particularly, Birrittella recites in the summary:

> The invention adds a matrix of flags or indicators to the architect registers for recording whether any overlapped instructions would modify the contents of any register that another of these instructions would require if it would have to be reexecuted. If it would, then that instruction is stalled or halted until the vector memory-reference instruction completes successfully.

Assuming for the sake of argument that stalling all instructions of a program results in a dead lock situation where the processor would grind to a complete halt, the Advisory Action seems to assert that this is the basis for concluding that Birrittella inherently discloses instructions of first and second programs. However, it is well known that to establish inherency, extrinsic evidence must make clear that the missing descriptive matter is necessary present in the thing described in the reference and it would be so organized by persons in the ordinary skill. Inherency, however, may not be established by probabilities or possibilities. The mere fact that certain thing may result from a given set of circumstances is not sufficient. *In re Oelrich*, 666 F.2d 578, 581-82 (CCPA 1981). Applicant does not believe the advisory action has made clear that a second program is necessarily described in Birrittella and that it would be so organized by persons in the ordinary skill. Simply put, Birrittella does not make clear that a first program is operating when a second program attempts to access a vector buffer. Indeed, one of ordinary skill in the art would understand that the overlapped instructions referred to above would indicate the existence of a single program.

## CONCLUSION

In view of the remarks set forth herein, the application is believed to be in condition for allowance and a notice to that effect is solicited. Nonetheless, should any issues remain that might be subject to resolution through a telephonic interview, the Examiner is invited to telephone the undersigned at 512-439-5093.

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Mail Stop AF, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on 7/12/04.

Attorney for Applicant(s) — Date of Signature: 7/12/04

Respectfully submitted,

Eric A. Stephenson
Attorney for Applicant(s)
Reg. No. 38,321
(512) 439-5093 [Phone]
(512) 439-5099 [Fax]